

#4

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of

SUZUKI

Serial No. 09/717,270

Filed: November 22, 2000

For: ELECTRONIC CONTROL UNIT HAVING SINGLE NON-VOLATILE MEMORY FOR MULTIPLE CENTRAL PROCESSING UNITS AND DATA RETRIEVAL METHOD

* * * * *



Atty. Ref.: 2018-354

Group: 2152

Examiner:

RECEIVED
JUL 26 2001
Technology Center 2100

July 24, 2001

Assistant Commissioner for Patents
Washington, DC 20231

SUBMISSION OF PRIORITY DOCUMENTS

Sir:

It is respectfully requested that this application be given the benefit of the foreign filing date under the provisions of 35 U.S.C. §119 of the following, a certified copy of which is submitted herewith:

<u>Application No.</u>	<u>Country of Origin</u>	<u>Filed</u>
11-334277	JAPAN	25 November 1999
2000-325346	JAPAN	25 October 2000

Respectfully submitted,

NIXON & VANDERHYTE P.C.

By: *Pam S. Nixon*
Pam S. Nixon
Reg. No. 25,640

LSN:vc
1100 North Glebe Road, 8th Floor
Arlington, VA 22201-4714
Telephone: (703) 816-4000
Facsimile: (703) 816-4100



日本国特許庁

PATENT OFFICE
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application:

2000年10月25日

出願番号
Application Number:

特願2000-325346

出願人
Applicant (s):

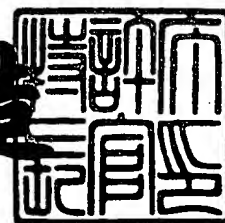
株式会社デンソー

CERTIFIED COPY OF
PRIORITY DOCUMENT

2000年12月 1日

特許庁長官
Commissioner,
Patent Office

及川耕造



出証番号 出証特2000-3099949

【書類名】 特許願

【整理番号】 PY20002160

【提出日】 平成12年10月25日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/167

【発明者】

【住所又は居所】 愛知県刈谷市昭和町1丁目1番地 株式会社デンソー内

【氏名】 鈴木 孝直

【特許出願人】

【識別番号】 000004260

【氏名又は名称】 株式会社デンソー

【代理人】

【識別番号】 100068755

【住所又は居所】 岐阜市大宮町2丁目12番地の1

【弁理士】

【氏名又は名称】 恩田 博宣

【電話番号】 058-265-1810

【選任した代理人】

【識別番号】 100105957

【住所又は居所】 東京都渋谷区代々木二丁目10番4号 新宿辻ビル8階

【弁理士】

【氏名又は名称】 恩田 誠

【電話番号】 03-5365-3057

【先の出願に基づく優先権主張】

【出願番号】 平成11年特許願第334277号

【出願日】 平成11年11月25日

【手数料の表示】

【予納台帳番号】 002956

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9908214

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 電子制御装置

【特許請求の範囲】

【請求項 1】 複数の CPU と、各 CPU で共用される不揮発性メモリとを備え、各 CPU 及び不揮発性メモリの間が相互に通信可能に接続される電子制御装置であって、

前記複数の CPU のうち、任意の CPU が不揮発性メモリにアクセスする前に、他の CPU によるアクセス中でないことを確認して、当該他の CPU に対して不揮発性メモリへアクセスする旨を通知し、不揮発性メモリへのアクセス終了後、他の CPU への通知を解除することを特徴とする電子制御装置。

【請求項 2】 各 CPU の間がポートを介して接続されており、任意の CPU が不揮発性メモリにアクセスする際、前記ポートの論理レベルをアクセス中を示す状態とする請求項 1 に記載の電子制御装置。

【請求項 3】 電源投入時における初期化処理を実施する際、各 CPU が不揮発性メモリにアクセスするタイミングをずらす請求項 1 又は 2 に記載の電子制御装置。

【請求項 4】 請求項 3 に記載の電子制御装置において、

不揮発性メモリへのアクセスのタイミングをずらすべくシステムレジスタの初期化処理を分割して行う電子制御装置。

【請求項 5】 請求項 3 又は 4 に記載の電子制御装置において、

初期化処理を実施する際、不揮発性メモリに格納されるデータのうち、予め区分けされた優先度の高いデータのみを読み出し、優先度の低いデータはその後、制御に必要になると読み出すこととする電子制御装置。

【請求項 6】 前記複数の CPU のうち、何れか一つの CPU がデータ読み出し要求を発行し、その要求に従い、各 CPU で共通に使用するデータを前記不揮発性メモリから読み出して各 CPU で同時に受信する請求項 1 に記載の電子制御装置。

【請求項 7】 請求項 6 に記載の電子制御装置において、

前記一つの CPU がデータ読み出し要求を発行する際、他の CPU では、該読

み出し要求を受信する電子制御装置。

【請求項 8】 請求項 6 又は 7 に記載の電子制御装置において、
前記複数の CPU では、事前に同期を取り不揮発性メモリからの共通使用データの読み取りを開始する電子制御装置。

【請求項 9】 請求項 6 ～ 8 の何れかに記載の電子制御装置において、
前記共通使用データの読み出し後、各 CPU では個別データの読み出しを順次実施する電子制御装置。

【請求項 1 0】 請求項 6 ～ 9 の何れかに記載の電子制御装置において、
前記複数の CPU のうち、一つをマスタ CPU、他をスレーブ CPU とし、マスタ CPU がデータ読み出し要求を発行し、該要求に応じて読み出された全データをマスタ CPU 及びスレーブ CPU で受信する電子制御装置。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、複数の CPU を持つ電子制御装置において、1 つの不揮発性メモリを共有する技術に関する。

【 0 0 0 2 】

【従来の技術】

近年、例えば車載されることを用途とする電子制御装置では、制御システムの高機能化や高信頼性化の要求に伴い、複数の CPU を使用した構成が多用されつつある。また、電子制御装置は、バッテリーが外されても記憶内容を保持する不揮発性メモリとして EEPROM 等を備え、その EEPROM には、仕向け地、MT（手動変速機）／AT（自動変速機）、過給器の有無等の車両情報や故障情報のデータが記憶される。

【 0 0 0 3 】

複数の CPU を搭載した電子制御装置として、CPU 毎に EEPROM をそれぞれ設け、各々の CPU で個別の EEPROM にアクセスしてデータの読み出し／書き込みを行うものがある。ところが、かかる装置では、使用する CPU と同じ数だけ EEPROM が必要となり、コストアップの原因となる。

【 0 0 0 4 】

これに対して、特開昭 6 2 - 1 3 9 0 6 4 号公報に開示された「マルチコンピュータシステム」では、複数の CPU のうち、何れか一つにのみ EEPROM を接続し、この EEPROM を各 CPU で共用することにより、EEPROM の必要数を減らしコスト削減を図っている。

【 0 0 0 5 】

より詳しくは、図 8 に示されるように、電子制御装置 2 0 は、2 つの CPU 2 1, 2 2 と、1 つの EEPROM 2 3 とを備える。CPU 2 1, 2 2 間は通信ライン 2 4, 2 5 を介して接続されている。また、一方の CPU 2 1 は信号ライン 2 6 を介して EEPROM 2 3 に接続されている。本装置において、例えばソフトウェアの初期化処理時には、CPU 2 1 は EEPROM 2 3 からデータを読み出し、更にそのデータを CPU 間の通信ライン 2 4 を介して CPU 2 2 に送信する。これにより、各 CPU 2 1, 2 2 で EEPROM 2 3 のデータが使用可能となる。また、CPU 2 2 のデータを EEPROM 2 3 へ書き込む場合には、書き込むべき全データを CPU 間の通信ライン 2 5 を介して CPU 2 1 へ送信し、CPU 2 1 が EEPROM 2 3 へのデータ書き込みを行う。

【 0 0 0 6 】

【発明が解決しようとする課題】

しかしながら、上記図 8 の装置においては、CPU 2 1 が仲介となって EEPROM 2 3 がアクセスされるので、CPU 2 2 ではアクセス時間が増大する、或いは CPU 間の通信データ量が増大するといった問題が生ずる。すなわち、ソフトウェアの初期化処理時には、一方の CPU 2 1 が EEPROM データを読み出し、他方の CPU 2 2 では、CPU 2 1 で読み出したデータを受信した後でなければ使用することができない。それ故、読み出したデータを処理に反映するのが遅れてしまう。また、CPU 2 2 が EEPROM 2 3 へデータを書き込む場合、その全データを一旦 CPU 2 1 へ送信しなくてはならないので、CPU 2 1, 2 2 間の通信データ量が膨大になる。

【 0 0 0 7 】

本発明の目的は、複数の CPU で一つの不揮発性メモリを共用する電子制御装

置において、不揮発性メモリへ効率良くアクセスすると共に、CPU間の通信データ量を削減することである。

【0008】

【課題を解決するための手段】

請求項1に記載の電子制御装置によれば、何れかのCPUが不揮発性メモリとアクセスする際、他のCPUによるアクセス中でないことが事前に確認された後、不揮発性メモリのデータ読み出し又は書き込みが行われる。また、このアクセス時には、その旨が他のCPUに通知される。従って、コスト削減を図るべく、複数のCPUで一つの不揮発性メモリを共用する電子制御装置において、特定のCPUを介して不揮発性メモリへのアクセスが行われる従来装置（図8の装置）に比べて、不揮発性メモリへのアクセス時間を大幅に短縮することができる。また、不揮発性メモリへのアクセス時には、その旨を該当するCPUから他のCPUに対して通知するだけでよいので、不揮発性メモリへの書き込みデータ等をCPU間で通信する従来装置（図8の装置）に比べ、CPU間の通信データ量が大幅に削減できる。

【0009】

請求項2に記載の発明では、任意のCPUが不揮発性メモリにアクセスする際、ポートの論理レベルをアクセス中を示す状態とする。要するに、アクセス中でない他のCPUは、ポートの論理レベルにより不揮発性メモリへのアクセスの有無、すなわちアクセスの可否を判断する。この場合、電子制御装置の簡易構成が実現できる。

【0010】

また、請求項3に記載の発明によれば、電源投入時における初期化処理を実施する際、各CPUが不揮発性メモリにアクセスするタイミングをずらすので、電源投入当初において資源利用の競合によるアクセス待ちが起こることはなく、そのアクセス待ちに起因して初期化時間が長引くことが防止される。それ故、初期化時間の短縮を実現することができる。

【0011】

この場合、請求項4に記載したように、不揮発性メモリへのアクセスのタイミ

ングをずらすべくシステムレジスタの初期化処理を分割して行うと良い。実際には、1つのCPUを除き、それ以外のCPUについて、不揮発性メモリへのアクセス時期が重複しないようシステムレジスタの初期化処理を分割して設定しておく。

【0012】

更に、請求項5に記載の発明によれば、初期化処理を実施する際、不揮発性メモリに格納されるデータのうち、予め区分けされた優先度の高いデータのみを読み出し、優先度の低いデータはその後、制御が必要になると読み出すので、電源投入の当初には必要最小限のデータだけが読み出されることとなる。従って、各々のCPUにおける初期化処理が完了するまでの時間が短縮される。特に、車両の仕向け情報や車両の仕様を表すデータについて優先度を高くしておけば、電子制御装置の起動に伴い、その当初から車両の走行状態や車載エンジンの運転状態が好適に制御できるようになる。

【0013】

また、請求項6に記載の発明によれば、複数のCPUのうち、何れか一つのCPUがデータ読み出し要求を発行し、その要求に従い、各CPUで共通に使用するデータを前記不揮発性メモリから読み出して各CPUで同時に受信する。それ故、各CPUで使用する共通データの読み出しが1回だけで済み、各CPUでデータを個々に読み出す場合に比べて初期化時間の短縮を図ることができる。

【0014】

上記請求項6の発明では、請求項7に記載したように、前記一つのCPUがデータ読み出し要求を発行する際、他のCPUでは、該読み出し要求を受信すると良い。これにより、読み出し要求を発行するCPUとは異なる他のCPUにおいても、共通データとして読み出されるデータのアドレスを認識することができる。

【0015】

また、請求項8に記載したように、前記複数のCPUでは、事前に同期を取り不揮発性メモリからの共通使用データの読み取りを開始すると良い。この場合、各CPUで読み出しデータが同時に受信できることの確実性が増し、ひいては不

揮発性メモリのデータ読み出しがより一層好適に実施できるようになる。

【0016】

また、請求項9に記載したように、前記共通使用データの読み出し後、各CPUでは個別データの読み出しを順次実施すると良く、これにより、各CPUでイニシャル時に必要とする全データの読み出しが完了する。

【0017】

更に、請求項10に記載したように、複数のCPUのうち、一つをマスタCPU、他をスレーブCPUとし、マスタCPUがデータ読み出し要求を発行し、該要求に応じて読み出された全データをマスタCPU及びスレーブCPUで受信すると良い。この場合、マスタCPUの要求に伴い全CPUで共通使用データの読み出しが行われるようになる。

【0018】

【発明の実施の形態】

（第1の実施の形態）

以下、この発明を具体化した第1の実施の形態を図面に従って説明する。本実施の形態の電子制御装置（ECU）は、複数のCPUを備えるマルチCPUシステムを構築するものであり、車両の走行状態やエンジンの運転状態を制御する。本実施の形態の装置は特に、2個のCPUで1個のEEPROMを共有するシステムを実現する。

【0019】

図1は、ECUの概要を示すブロック図である。同図において、ECU10は、第1CPU11と第2CPU12とEEPROM13とを備え、各CPU11及び12間は汎用ポート14、15を介して接続されている。また、各CPU11、12とEEPROM13は、データ読み出し／書き込み用のチップセレクトライン16、クロックライン17、SCI（Serial Communication Interface）ライン18をそれぞれ共有し、各ラインを介して相互に接続されている。なお本実施の形態では、汎用ポート14、15の能動レベル（Active level）をLレベルとし、そのLレベルの状態をEEPROM使用中と定義する。

【0020】

次に、ECU10内の各CPU11, 12により実施される各種演算処理について図2～図6のフローチャートを参照しながら順に説明する。

先ずは図2のフローチャートを用い、ソフトウェアの初期化時におけるEEPROM13のデータ読み出し処理を説明する。ECU10に電源が投入されると、第1CPU11及び第2CPU12はほぼ同時に起動し、第1CPU11は図2(a)のメインルーチンで初期化処理を実行し、第2CPU12は図2(b)のメインルーチンで初期化処理を実行する。

【0021】

詳細には、第1CPU11が起動すると、図2(a)のステップ101では全てのシステムレジスタのうち、EEPROM13にアクセスするために必要なシステムレジスタだけを初期化し、続くステップ102では、EEPROM13内の重要データを読み出す。このとき、EEPROM13の全データのうち、仕向け地、MT/AT、過給器の有無、VINコード等の車両情報や故障情報のデータなど、車両制御に大きな影響がある比較的重要なデータのみを読み出す。こうして重要データのみを優先して初期化することにより、起動直後の初期化処理が必要以上に長引かず、EEPROM13へのアクセス時間が短縮される。

【0022】

次に、ステップ103では、残りのシステムレジスタの初期化を行い、続くステップ104では、アプリケーションデータの初期化を行う。このアプリケーションデータの初期化時には、既にEEPROMデータが読み出されており、EEPROMデータを用いてアプリケーション処理が適宜切り替えられる。このステップ104までが起動当初における一連の初期化処理に相当する。

【0023】

その後、通常の各種制御が実施される途中で、EEPROM13の残りのデータが必要になると、該当するデータがEEPROM13から適宜読み出される（例えば、ステップ110）。

【0024】

一方、第2CPU12が起動すると、先ず図2(b)のステップ201では、システムレジスタの初期化を行う。このとき、前記図2(a)の処理では、シス

テムレジスタの初期化処理を前後 2 つに分割して実施したのに対し、図 2 (b) の処理では、全てのシステムレジスタの初期化処理をまとめて実施する。

【 0 0 2 5 】

続くステップ 2 0 2 では、前記図 2 (a) のステップ 1 0 2 と同様に、仕向け地、MT / AT、過給器の有無、VIN コード等の車両情報や故障情報のデータなど、車両制御に大きな影響がある比較的重要なデータのみを E E P R O M 1 3 から読み出す。次に、ステップ 2 0 3 では、アプリケーションデータの初期化を行う。

【 0 0 2 6 】

その後、図 2 (a) と同様に、通常の各種制御が実施される途中で、E E P R O M 1 3 の残りのデータが必要になると、該当するデータが E E P R O M 1 3 から適宜読み出される（例えば、ステップ 2 1 0）。

【 0 0 2 7 】

図 2 (a) , (b) を比較すると前述の通り、前者はシステムレジスタの初期化処理を分割して行い、後者はシステムレジスタの初期化を一括して行うので、E E P R O M 1 3 へのアクセスの時期がずらされることとなる。それ故、各 C P U 1 1 , 1 2 において E E P R O M 1 3 へのアクセス待ちが生じることはなく、初期化処理が効率良く実施される。

【 0 0 2 8 】

次に、E E P R O M 1 3 のデータ読出処理の流れを、図 3 及び図 4 のフローチャートを参照しながら説明する。なお、以下の説明では、第 1 C P U 1 1 によりデータ読み出しが実施される場合を例にして説明する。

【 0 0 2 9 】

図 3 の処理は、例えば、前記図 2 (a) のステップ 1 0 2 , 1 1 0 の処理や図 2 (b) のステップ 2 0 2 , 2 1 0 の処理にて起動される他、時間同期処理等から E E P R O M 読み出し要求がコールされた場合に起動される。特にこの図 3 の処理では、図 2 (a) のステップ 1 1 0 や図 2 (b) のステップ 2 1 0 において E E P R O M 1 3 へのアクセスが重複する場合にそれを調停する。

【 0 0 3 0 】

最初にステップ301では、他側のCPU（この場合は第2CPU12）がEEPROM13を使用中でないか否かを判別する。これは、第2CPU12に接続された汎用ポート15がLレベルかどうかで判断する。EEPROM使用中の場合はステップ307へジャンプし、リターンコードRCを「ビジー（Busy）」とする。すなわち、EEPROMデータが読み出せなかったことを上位関数に伝える。

【0031】

EEPROM未使用の場合は、ステップ302に進み、汎用ポート14をLレベルとして第2CPU12に対してEEPROM使用中である旨を通知する。続くステップ303では、読み出しコマンドを送信レジスタにセットする。その後、ステップ304では処理モードを「読み出し」にセットし、ステップ305では受信割り込みを許可する。更に、ステップ306では、リターンコードRCを「OK」としてEEPROMデータが読み出せることを上位関数に伝え、本処理を終了する。

【0032】

続いて、EEPROM13からのデータ受信割り込みを受信すると、第1CPU11は図4の処理を起動する。図4において、始めにステップ401では他の受信割り込みを禁止する。続くステップ402では、EEPROM13からの受信データを取得し、その受信データをレジスタから第1CPU11内のRAMへコピーする。

【0033】

その後、ステップ403では、処理モードを「無し（None）」に戻す。また、続くステップ404では、汎用ポート14をHレベルに戻し、第2CPU12に対してEEPROM未使用とした旨を通知する。

【0034】

なお、ECU10の初期化に際して図3の処理が起動される場合、すなわち図2（a）、（b）のステップ102、202で図3の処理が起動される場合には、同図のステップ301が必ずYESとなり、後続のEEPROM読み出し処理が直ちに実行される。

【 0 0 3 5 】

次に、EEPROM13へのデータ書込処理の流れを、図5及び図6のフローチャートを用いて説明する。なお、ここでも図3及び図4での説明と同様に、第1CPU11によりデータ書き込みが実施される場合を例にして説明する。

【 0 0 3 6 】

図5の処理は、例えば、時間同期処理等からEEPROM書き込み要求がコールされた場合に起動される。最初にステップ501では、汎用ポート15の状態に基づいて他側のCPU（この場合は第2CPU12）がEEPROM13を使用中でないか否かを判別する。EEPROM使用中の場合はステップ506へジャンプし、リターンコードRCを「ビジー」とする。

【 0 0 3 7 】

EEPROM未使用の場合は、ステップ502に進み、汎用ポート14をLレベルとして第2CPU12に対してEEPROM使用中である旨を通知する。続くステップ503では、書き込みコマンドを送信レジスタにセットする。その後、ステップ504では処理モードを「書き込み」にセットし、ステップ505ではリターンコードRCを「OK」として本処理を終了する。

【 0 0 3 8 】

一方、所定の時間周期で割り込みが発生すると、第1CPU11は図6の時間同期割り込み処理を起動する。図6において、ステップ601では、処理モードが「書き込み」になっていることを確かめ、続くステップ602では、EEPROM13がビジー状態（書き込みコマンド処理中状態）でないことを確かめる。処理モードが書き込みでない場合、或いはEEPROMビジーの場合（ステップ601がNO、或いはステップ602がYESの場合）、そのまま本処理を終了する。

【 0 0 3 9 】

また、ステップ603に進むと、処理モードを「無し（None）」に戻す。また、続くステップ604では、汎用ポート14をHレベルに戻し、第2CPU12に対してEEPROM未使用とした旨を通知する。

【 0 0 4 0 】

要するに、上述した図 3～図 6 が本発明の要旨にかかる部分であり、例えば第 1 CPU 1 1 が EEPROM 1 3 とアクセス（読み出し又は書き込み）する前に、第 2 CPU 1 2 によるアクセス中でないことが事前に確認され（ステップ 3 0 1, 5 0 1）、アクセス中でなければ、EEPROM 1 3 へアクセスする旨が第 2 CPU 1 2 に通知される（ステップ 3 0 2, 5 0 2）。また、EEPROM 1 3 へのアクセス終了後、アクセス中である旨の通知が解除される（ステップ 4 0 4, 6 0 4）。第 2 CPU 1 2 がこれら図 3～図 6 の処理を実施する場合も同様である。

【0041】

かかる場合、各 CPU 1 1, 1 2 はそれぞれ、EEPROM 1 3 に直接アクセスし、読み出し又は書き込みを行う。つまり、従来装置（例えば図 8 の装置）とは異なり、CPU 間通信を介さなくても個々の CPU 1 1, 1 2 で EEPROM データが授受される。またこのとき、CPU 毎に EEPROM を個別に持つ構成と比較しても、CPU 間の通信データの量が増えることはない。

【0042】

以上詳述した本実施の形態によれば、以下に示す効果が得られる。

（イ）コスト削減を図るべく第 1 及び第 2 CPU 1 1, 1 2 で一つの EEPROM 1 3 を共用する ECU 1 0 において、EEPROM 1 3 へのアクセス時間を大幅に短縮することができると共に、CPU 1 1, 1 2 間の通信データ量が大幅に削減できる。

【0043】

（ロ）汎用ポート 1 4, 1 5 の論理レベルにて EEPROM 1 3 へのアクセスの有無が通知されて、アクセスの可否が判断されるので、ECU 1 0 の簡易構成が実現できる。

【0044】

（ハ）ECU 1 0 の初期化に際し、各 CPU 1 1, 1 2 が EEPROM 1 3 にアクセスするタイミングをずらしたので、EEPROM 資源利用の競合によるアクセス待ちが起こることはなく、そのアクセス待ちに起因して初期化時間が長引くことが防止される。それ故、初期化時間の短縮を実現することができる。この

場合、CPU毎にEEPROMを持つ構成と比較しても、初期化時間はほぼ同等である。

【0045】

(二) ECU10の初期化に際し、車両情報や故障情報のデータなど車両制御に大きな影響がある比較的重要なデータ（優先度の高いデータ）のみを読み出し、それ以外のデータ（優先度の低いデータ）はその後、制御に必要になると読み出すので、ECU10への電源投入当初には、必要最小限のデータだけが読み出される。従って、各々のCPU11, 12における初期化処理が完了するまでの時間が短縮される。この場合、ECU10の起動当初から車両の走行状態や車載エンジンの運転状態が好適に制御できるようになる。

【0046】

(ホ) アプリケーションデータの初期化よりも前にEEPROMデータを読み出すため、アプリケーション処理中にEEPROMデータを参照しながら処理の切り替えを実施することが可能となる。

【0047】

(第2の実施の形態)

本実施の形態では、電源投入時の初期化処理（イニシャル処理）において、各CPUで共通使用するデータをEEPROMから同時に読み込み、初期化時間の短縮を図るようにしたECUについて説明する。以下、上記第1の実施の形態との相違点のみを説明する。なお、上記図1の構成において、第1CPU11をマスターCPU、第2CPU12をスレーブCPUとする。

【0048】

図7(a), (b)は、第1及び第2CPU11, 12における初期化処理を示すフローチャートであり、ECU10に電源が投入されると、各CPU11, 12においてほぼ同時に図7(a)及び(b)の処理がそれぞれ起動される。

【0049】

図7(a)がスタートすると、先ずステップ701では、システムレジスタの初期化を実施する。続いてステップ702では、第1及び第2CPU11, 12で共通使用するデータ領域の読み出しコマンドを、前記図1のSCIライン18

を介してEEPROM13と第2CPU12に対して送信する。その後、ステップ703では、前記読み出しコマンドに応答してEEPROM13から返信される読み出しデータをSCIライン18を介して受信する。以降、残りの共通データについても前述のステップ702、703と同様に、読み出しコマンドの送信、並びにEEPROM読み出しデータの受信を繰り返し実施する（ステップ704、705・・・）。

【0050】

そして共通データの読み出しが終了すると、ステップ711では、当該第1CPU11で個別に使用するデータを読み出す。その後、ステップ712では、アプリケーションデータの初期化を実施し、更に、制御途中のステップ713では、当該制御に必要なEEPROMデータの読み出しを実施する。

【0051】

一方、図7（b）がスタートすると、先ずステップ801では、前記図7（a）のステップ701と同様に、システムレジスタの初期化を実施する。続いてステップ802では、前記図7（a）のステップ702で第1CPU11が発行したEEPROMデータ（共通データ）の読み出しコマンドをSCIライン18を介して受信する。その後、ステップ803では、前記読み出しコマンドに応答してEEPROM13から返信される読み出しデータをSCIライン18を介して受信する。以降、残りの共通データについても前述のステップ802、803と同様に、読み出しコマンドの受信、並びにEEPROM読み出しデータの受信を繰り返し実施する（ステップ804、805・・・）。

【0052】

そして共通データの読み出しが終了すると、ステップ811では、当該第2CPU12で個別に使用するデータを読み出す。その後、ステップ812では、アプリケーションデータの初期化を実施し、更に、制御途中のステップ813では、当該制御に必要なEEPROMデータの読み出しを実施する。

【0053】

上記図7の処理によれば、第1CPU（マスタCPU）11により読み出しコマンドが発行された後、EEPROM13から送信されてくるデータが各CPU

1 1, 1 2で同時に受信される。この際、第2 CPU (スレーブCPU) 1 2ではSCIライン1 8を介して読み出しコマンドを受信しているので、共通データのどの部分 (アドレス) の読み出しコマンドであるかが認識される。そして、共通使用データの読み出し後、個別データの読み出しが順次実施されると、各CPU 1 1, 1 2でイニシャル時に必要とする全データの読み出しが完了する。各CPU 1 1, 1 2では、アドレス情報に基づいて受信データが対応するRAMへ各々コピーされ、必要に応じて使用される。

【0 0 5 4】

以上本実施の形態によれば、各CPU 1 1, 1 2で使用する共通データの読み出しが1回だけで済み、各CPU 1 1, 1 2でデータを個々に読み出す場合に比べて初期化時間の短縮を図ることができる。また、電源投入直後においてシステムレジスタの初期化に引き続いて共通使用データの読み出しを開始することにより、各CPU 1 1, 1 2でのデータ読み取りのタイミングを概ね合わせる (同期を取る) ことが可能となる。これにより、各CPU 1 1, 1 2で読み出しデータが同時に受信できることの確実性が増し、ひいてはEEPROM 1 3のデータ読み出しがより一層好適に実施できるようになる。

【0 0 5 5】

なお本発明は、上記以外に次の形態にて具体化できる。

上記第1の実施の形態では、EEPROM使用中か否かを汎用ポート1 4, 1 5の論理レベルで判定したが、これを変更する。例えば、第1及び第2 CPU 1 1, 1 2間をポートで接続せず、チップセレクトの状態をチェックすることによりEEPROM使用中か否かを判断する。この場合、ECU 1 0の構成をより一層簡略化できる。

【0 0 5 6】

ECU 1 0内に3つ以上のCPUを設ける場合にも本発明は勿論適用できる。この場合、図2の初期化処理では、EEPROMデータを読み出すタイミングが互い違いになるよう、システムレジスタの初期化処理やそれにアプリケーションデータの初期化を加えた処理等を分割して設定しておくが良い。

【0 0 5 7】

上記第2の実施の形態では、各CPUでのシステムレジスタ初期化に引き続いて共通使用データの読み出しを開始し、それにより各CPUでの同期を取る構成としたが、各CPUで動作周期が異なる場合等、同期が取りにくい場合には、チップセレクトライン16を用いて同期を取るようにしても良い。勿論、3つ以上のCPUで共通に使用されるデータをEEPROMから同時に受信するよう構成することも可能である。この場合、3つ以上のCPUのうち、何れか一つのCPUで読み出しコマンド（読み出し要求）を発行すればよい。

【0058】

本発明が適用できる不揮発性メモリとして、既述のEEPROM以外に、フラッシュメモリを用いることも可能である。

【図面の簡単な説明】

【図1】 発明の実施の形態におけるECUの概要を示す構成図。

【図2】 第1CPU及び第2CPUにおけるメインルーチンを示すフローチャート。

【図3】 読み出し要求処理を示すフローチャート。

【図4】 受信割り込み処理を示すフローチャート。

【図5】 書き込み要求処理を示すフローチャート。

【図6】 時間同期処理を示すフローチャート。

【図7】 第2の実施の形態において各CPUでの初期化処理を示すフローチャート。

【図8】 従来技術におけるECUの構成を示す図。

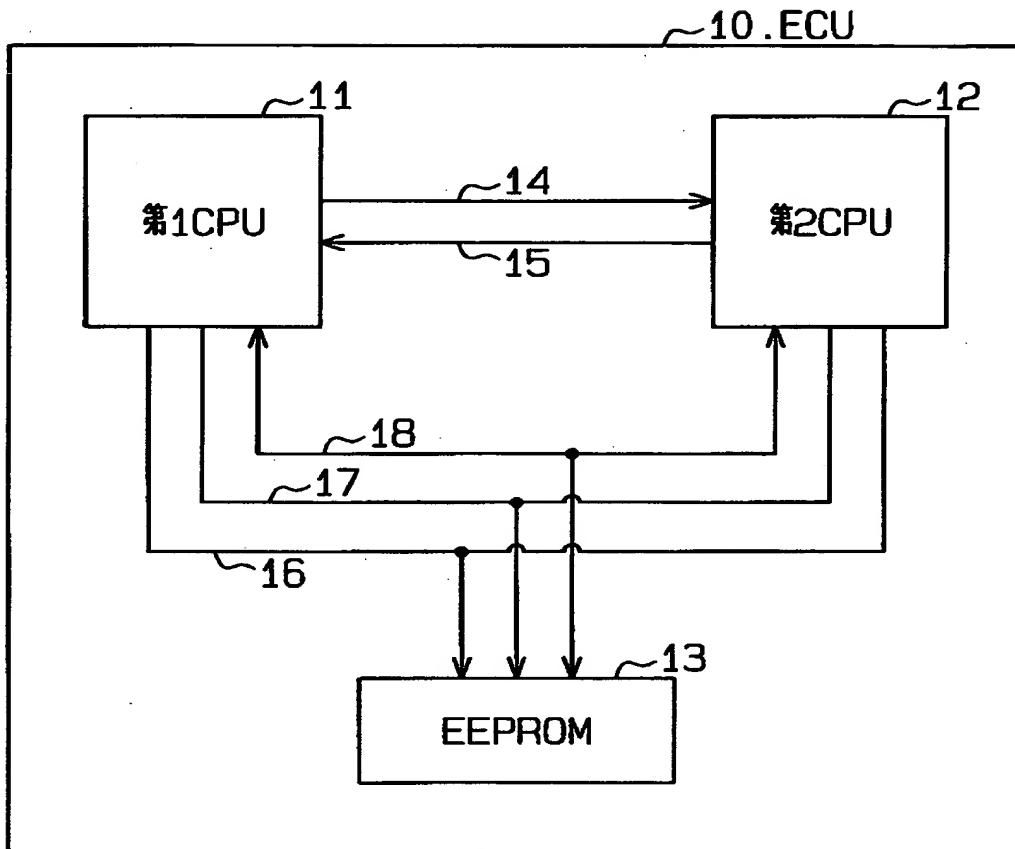
【符号の説明】

10…ECU、11…第1CPU、12…第2CPU、13…不揮発性メモリとしてのEEPROM、14、15…汎用ポート。

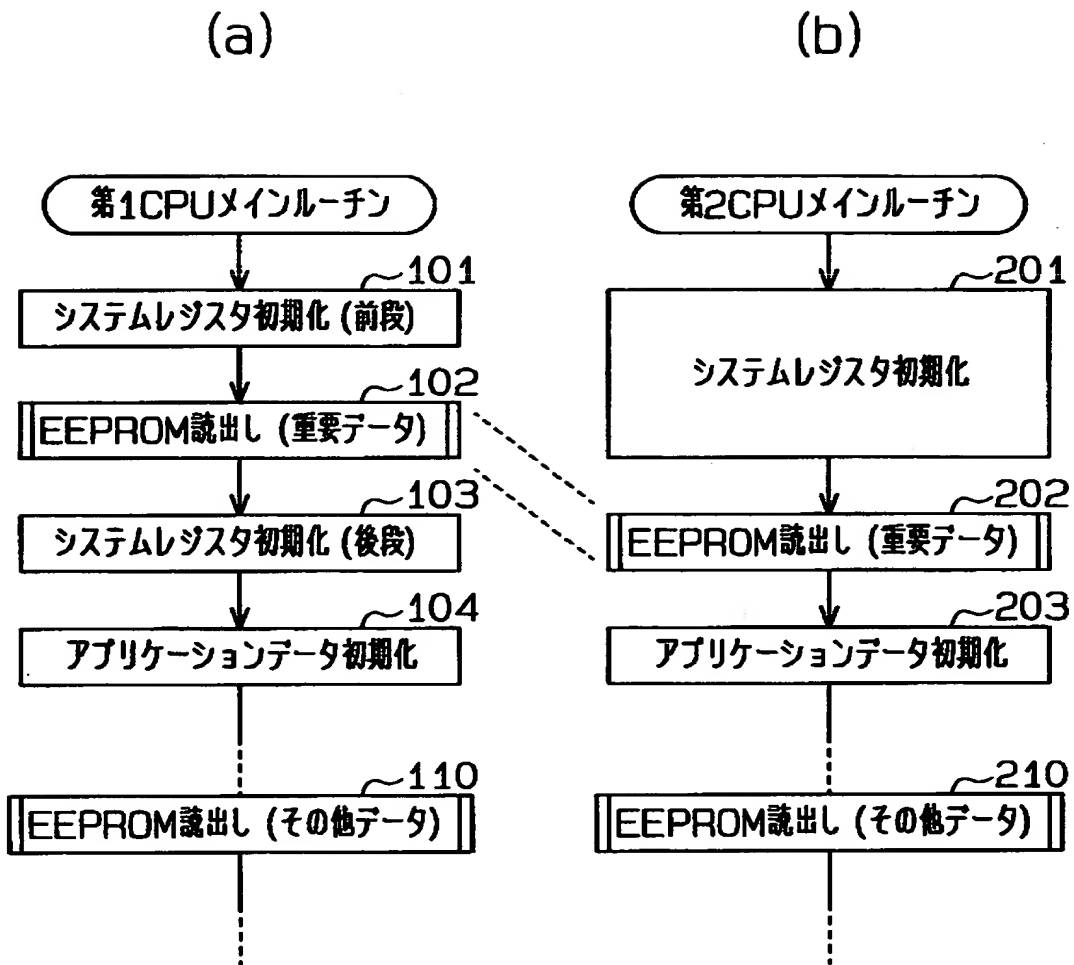
【書類名】

図面

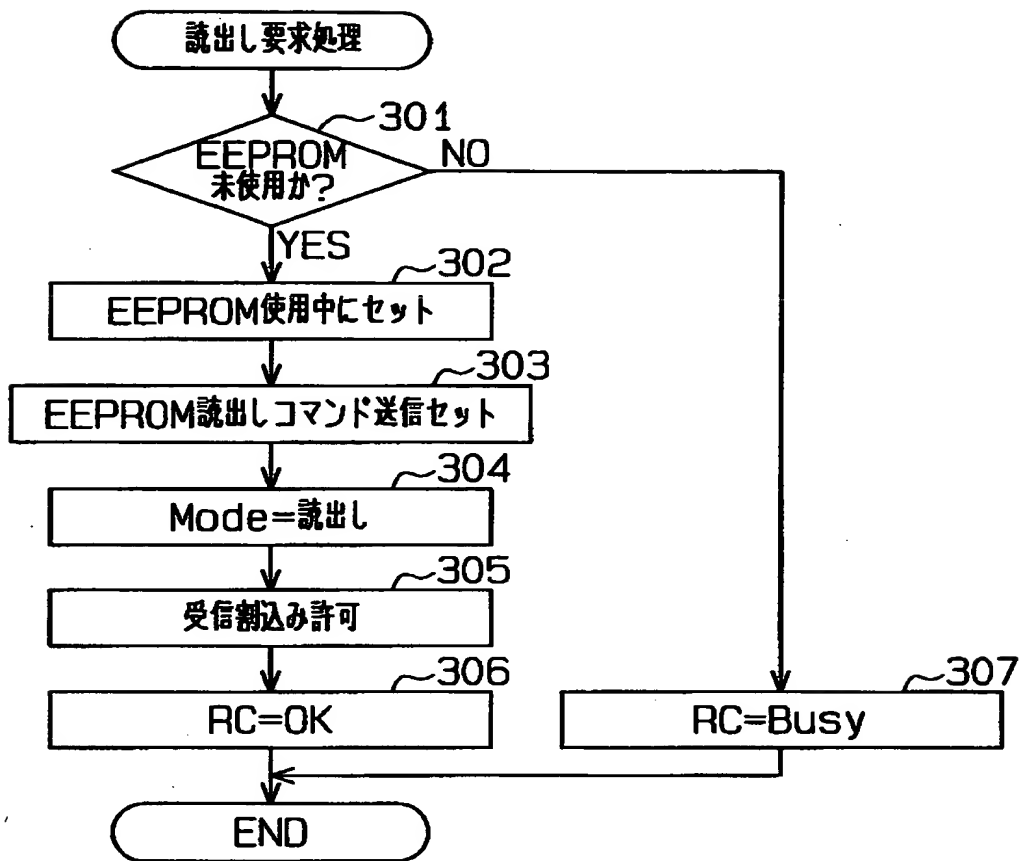
【図 1】



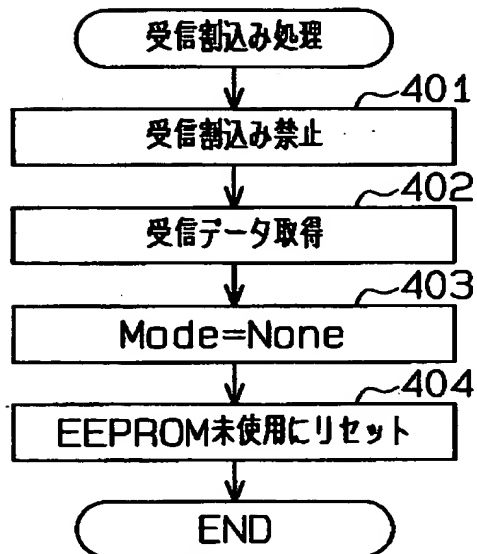
【図 2】



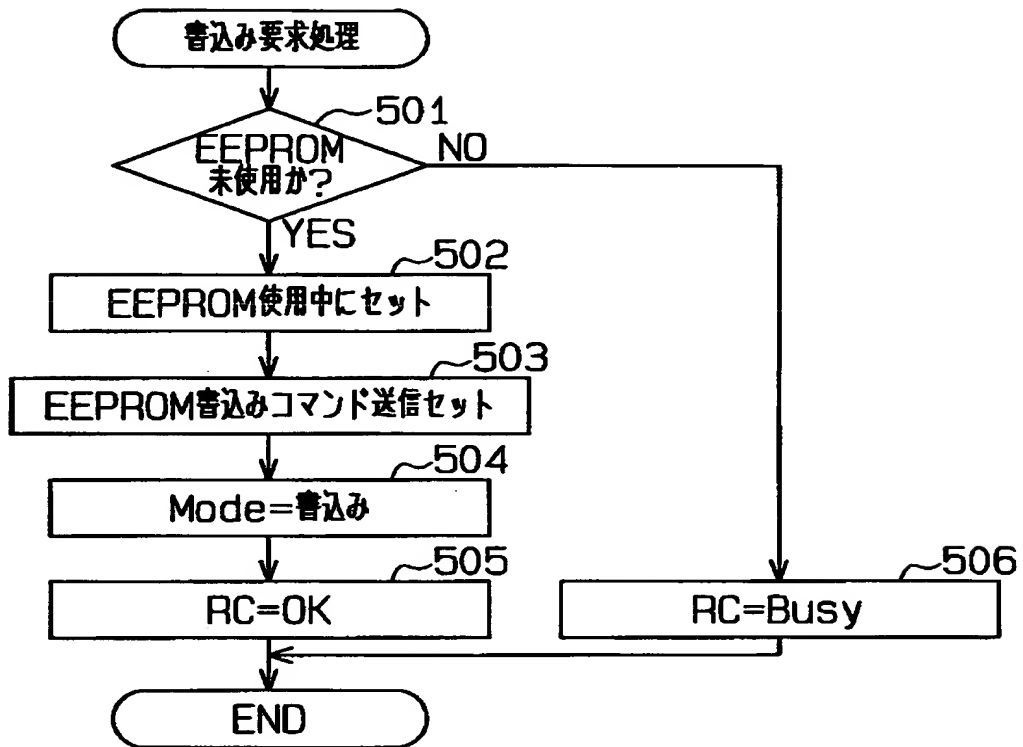
【図 3】



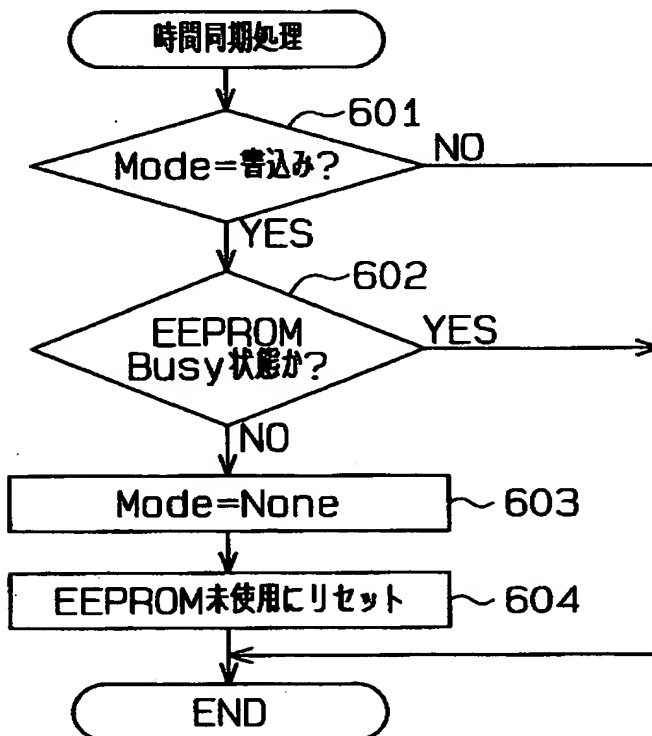
【図 4】



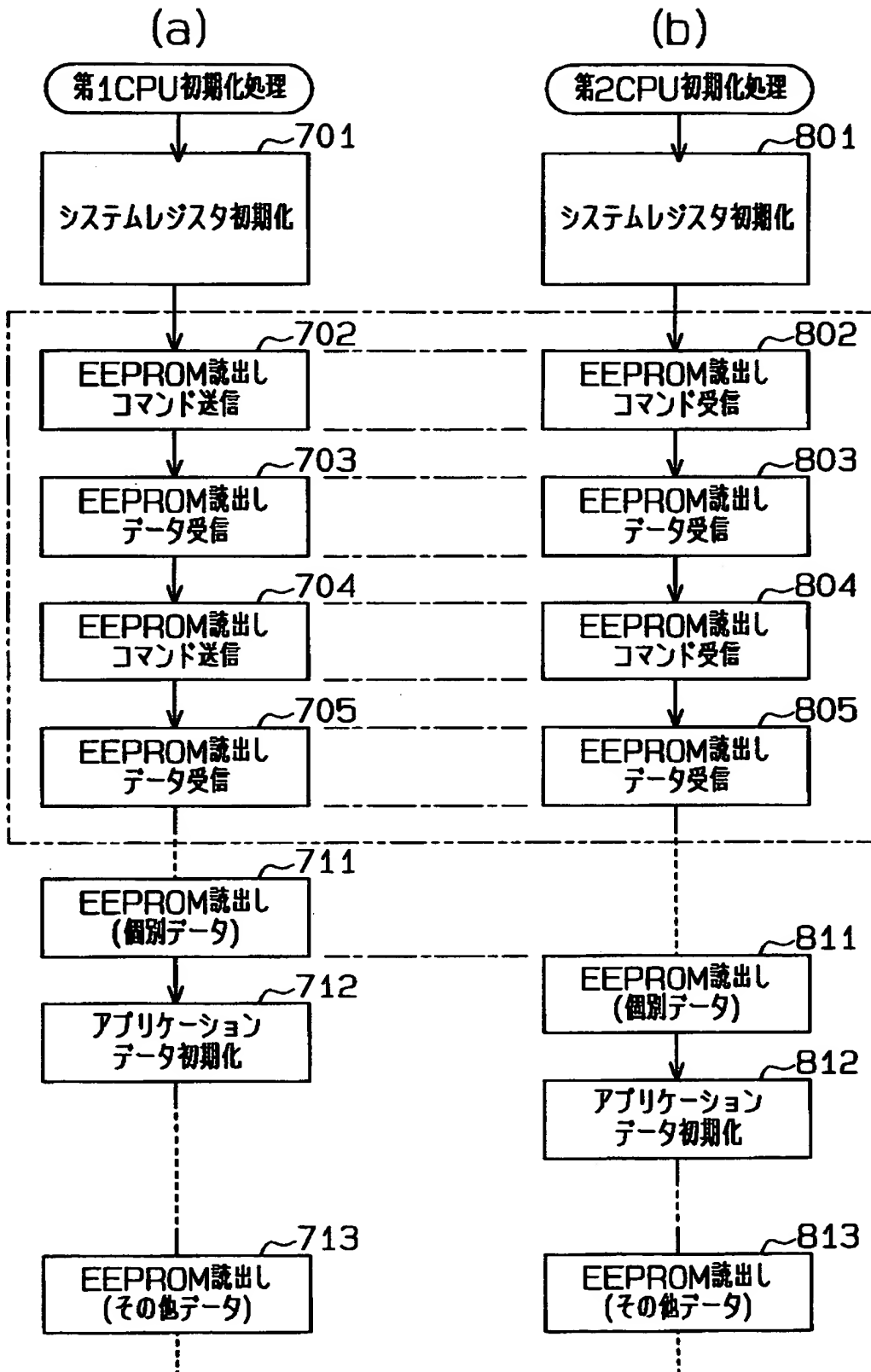
【図 5】



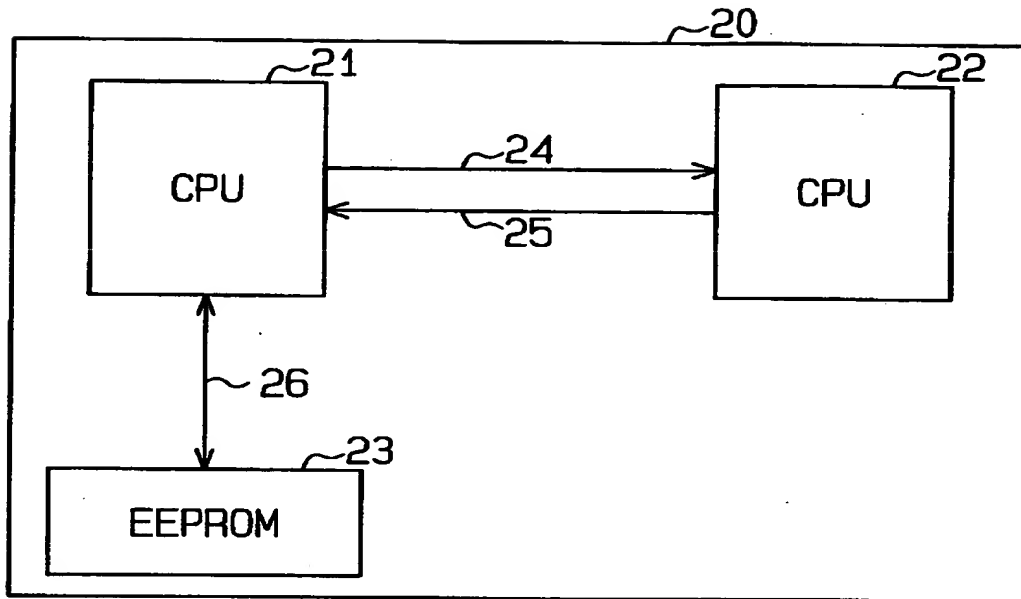
【図 6】



【図 7】



【図 8】



【書類名】 要約書

【要約】

【課題】複数のCPUで一つの不揮発性メモリを共用する電子制御装置において、不揮発性メモリへ効率良くアクセスすると共に、CPU間の通信データ量を削減する。

【解決手段】ECU10は、第1CPU11と第2CPU12とEEPROM13とを備え、各CPU11及び12間は汎用ポート14, 15を介して接続されている。各CPU11, 12とEEPROM13は、チップセレクトライン16、クロックライン17、SCIライン18をそれぞれ共有し、各ラインを介して相互に接続されている。例えば第1CPU11がEEPROM13とアクセス（読み出し又は書き込み）する前に、第2CPU12によるアクセス中でないことが事前に確認され、アクセス中でなければ、EEPROM13へアクセスする旨が第2CPU12に通知される。また、EEPROM13へのアクセス終了後、アクセス中である旨の通知が解除される。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号

[000004260]

1. 変更年月日	1996年10月 8日
[変更理由]	名称変更
住 所	愛知県刈谷市昭和町1丁目1番地
氏 名	株式会社デンソー